

بسمه تعالی

مجموعه آموزش‌های مجازی دانشگاه معین رشت

طراحی بانک‌های اطلاعاتی

جلسه دوم

مدرس سید علی اصغری

ارتباط با مدرس:

برای ارتباط با مدرس و طرح سؤالات خود می‌توانید از طرق زیر به ترتیب اولویت اقدام نمایید.

saasghari@chmail.ir

saasghari@gmail.com

(فقط واتس اپ) ۰۹۱۱۲۳۲۹۰۱۱

آدرس ایمیل اول (چاپار) برای امور آموزش و ارتباط با دانشجویان اختصاص داده شده است. در صورتیکه ایمیلتان به این آدرس ارسال نشد از آدرس دوم (گوگل) استفاده نمایید. لطفاً در هر ایمیلی که ارسال می‌کنید خودتان را بطور کامل معرفی کنید و علاوه بر نام و نام خانوادگی و شماره دانشجویی، نام دانشگاه و عنوان درس خود را حتماً بنویسید تا اشتباهی پیش نیاید.

تمرین‌هایی در لا به لای درس قرار داده می‌شود که می‌تواند در روند یادگیری کمک بزرگی باشد. حتماً این تمرین‌ها را حل کرده و جواب آنها را به آدرس ایمیل اول ارسال نمایید. می‌توانید جواب را روی یک کاغذ بنویسید، از آن عکس بگیرید و عکس را ارسال کنید. اگر تایپ می‌کنید حتماً آن را به pdf تبدیل کرده و هم فایل word و هم pdf را ارسال نمایید.

پایگاه داده رابطه‌ای:

در پایگاه داده رابطه‌ای، داده‌ها در داخل یک سری جدول‌ها سازماندهی می‌شوند. طراح پایگاه داده و برنامه‌نویس داده‌ها را در قالب جدول‌ها می‌بینند. اما احتمالاً این داده‌ها به همین شکل که دیده می‌شوند در حافظه ذخیره نشده‌اند. DBMS نرم‌افزاری است که فایل یا فایل‌های پایگاه داده را نگهداری کرده و داده‌ها را به صورت جدول‌هایی به طراح پایگاه داده نشان می‌دهد و جزئیات سطوح پایین و نحوه ذخیره سازی را از سطوح بالاتر مخفی می‌کند. SQL Server, DB2, MySQL, Oracle نمونه‌هایی از DBMS‌های رابطه‌ای هستند. مدل داده رابطه‌ای (سازماندهی داده‌ها در قالب یک سری جدول) بدلیل ملموس و ساده بودن پرکاربردترین مدل داده است. در این مدل هر جدول از تعدادی سطر و ستون تشکیل می‌شود. مانند مثال‌های صفحات بعد.

جدول course مربوط است به اطلاعات دروس

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

جدول instructor مربوط است به اطلاعات

مدرسين .

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

موجودیت: هر چیزی که دارای صفات و خواص

معینی باشد که میتوان به آنها مقادیری نسبت داد. به زبان ساده تر می توانیم بگوییم هر چیزی که وجود دارد یک موجودیت است. هر دانشجو، استاد، کارمند، هر یک از درس ها، کلاس ها و ... در دانشگاه، هر یک از فروشندگان، صندوق داران، هر یک از کالاها و هر یک از تولید کنندگان، هر مشتری و هر فاکتور و ... در یک فروشگاه نمونه هایی از موجودیت هستند.

فیلد: کوچکترین واحد داده‌های ذخیره شده است. داده‌ی مربوط به یکی از خصوصیت‌های یکی از موجودیت‌ها با نام فیلد شناخته می‌شود. در داخل جدول فیلدها در محل تلاقی سطرها و ستون‌ها قرار داده می‌شوند.

رکورد: مجموعه‌ای از فیلدهای مرتبط به هم. در واقع مجموعه فیلدهای مربوط به تمام صفات يك موجودیت رکورد نامیده می‌شود. در جدول هر رکورد در یک سطر از جدول قرار می‌گیرد.

جدول: هر جدول در پایگاه داده‌ی رابطه‌ای عبارت است از مجموعه‌ای از سطرها است که هر سطر از جدول در واقع يك رکورد است. هر جدول مجموعه‌ای از مقادیر صفات موجودیت‌های همسان است هر ستون از جدول شامل مقادیر یکی از صفات. به عنوان مثال جدول دروس که در آن اطلاعات درس‌ها ذخیره می‌شود (صفحه ۴). یا جدول مدرسین که اطلاعات مدرس‌ها در آن ذخیره می‌شود (صفحه ۵). در هر سطر اطلاعات یکی از مدرس‌ها ذخیره می‌شود.

دامنه: مجموعه مقادیر مجاز برای یک ستون از جدول یا یک صفت از موجودیت. مثل مجموعه تمام مقادیری که می‌توانند در ستون عنوان درس (title) نوشته شوند. می‌توان دامنه را معادل نوع داده برای متغیرها در برنامه‌نویسی دانست.

رابطه:

دو مجموعه A و B رابطه زیر در نظر بگیریم

$$A = \{a, b, c\} \quad B = \{n, y\}$$

ضرب دکارتی دو مجموعه، مجموعه‌ای است که از زوج‌های مرتبی که عناصر اول آن از مجموعه اول و عنصر دوم آن از مجموعه دوم انتساب شده باشد.

$$A \times B = \{(a, n), (b, n), (c, n), (a, y), (b, y), (c, y)\}$$

از نظر ریاضی رابطه به هر زیر مجموعه از ضرب دکارتی گفته می‌شود. مثل مجموعه‌های زیر

$$\{(a, n), (c, n), (a, y)\} \quad \text{خودمجموعه ضرب دکارتی } A \times B \text{ نیز زیرمجموعه}$$

خودش است. بنابراین خودی هم یک رابطه است.

$$\{(a, y), (b, n)\}$$

$$\{\} \quad \text{مجموعه تهی} \rightarrow \{\}$$

منظور از رابطه در اینجا دلیلی نیست.

اگر A و B را دامنه‌های مربوط به صفات یک مجموعه از موجوداتی همسان در نظر بگیریم می‌توانیم بگوییم که یک رابطه در اینجا دارد زیرمجموعه‌ای است از ضرب دکارتی دامنه‌ها.

هر رابطه را می توانیم به صورت مجموعه ای از زوج های مرتب
نمایش دهیم و هم به صورت جدول به این شکل توجه کنید

A	B
a	x
c	x
a	y

$$\leftarrow \{(a, x), (c, x), (a, y)\}$$

A	B
a	y
b	x

$$\leftarrow \{(a, y), (b, x)\}$$

A	B
c	x
b	y
c	y

$$\leftarrow \{(c, x), (b, y), (c, y)\}$$

سایه رنگی: سه دامنه D_1 ، D_2 و D_3 را در نظر بگیرید.

$D_1 = \{A-101, A-102, A-201, A-215, A-217, A-222, A-305, \dots\}$

D_1 مجموعه تمام شماره حسابی ممکن

$D_2 = \{\text{Downtown, Perryridge, Brighton, Mianus, } \dots\}$

D_2 مجموعه اسامی شعب

$D_3 = \mathbb{N}$

D_3 مجموعه موجودیهای حساب

تنوعی از این مجموعه نامهای زیر را در نظر بگیرید.

account_number	branch_name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750

بدلیل بزرگ بودن مجموعه نام

می توانیم حاصلضرب دکارتی آنرا

را بنویسیم اما می توانیم زیرمجموعه

حاصلضرب دکارتی را که یک رابطه

است بنویسیم.

تا اینجا حتماً متوجه ارتباط نزدیک و تشابه زیاد جدول و رابطه شده‌اید. همچنین نزدیکی و تشابه زیاد مفاهیم رکورد، سطر جدول، چندتایی مرتب یا tuple و موجودیت. با اینکه این مفاهیم با یکدیگر تفاوت دارند به دلیل شباهت و نزدیکی زیاد در مباحث پایگاه داده و حتی در کتاب‌ها بجای یکدیگر استفاده می‌شوند.

به عنوان نمونه‌ای از تفاوت‌ها از آنجاییکه رابطه یک مجموعه است بنابراین وجود عنصر (چندتایی مرتب) تکراری در آن معنی ندارد. اما در جدول می‌توانیم عنصر تکراری داشته باشیم. همچنین ترتیب عناصر در چندتایی مرتب مهم است اما در رکورد و در سطرهای جدول می‌توان ترتیب عناصر را تغییر داد.

ستون و صفت نیز اصطلاحاتی هستند که معادل یکدیگر بکار می‌روند. صفت مربوط است به موجودیت و ستون یکی از اجزای جدول است. بسیار پیش می‌آید که بجای ستون جدول از اصطلاح صفت جدول استفاده می‌کنیم. صفت‌ها در یک رابطه ممکن است دامنه یکسانی داشته باشند و از یک نوع باشند.

فیلدها ممکن است مقدار تهی داشته باشند. مقدار تهی به این معنی است که مقدار واقعی معلوم نیست یا وجود ندارد. مثلاً ممکن است شماره تلفن یک نفر را نداشته باشیم و در پایگاه داده ثبت نکنیم. در DBMS ها این امکان

وجود دارد که اجازه دهیم یک ستون مقدار تهی قبول کند یا نه. بدلیل مشکلاتی که مقادیر تهی در هنگام بازیابی داده‌ها و به‌هنگام‌سازی آنها بوجود می‌آورند بهتر است تا حد امکان از استفاده از مقادیر تهی اجتناب گردد.

شمای پایگاه داده:

در بحث پایگاه داده لازم است بین دو مفهوم مهم تمایز قائل شویم. یکی شمای پایگاه داده (database schema) است و دیگری نمونه پایگاه داده (database instance). شمای پایگاه داده طراحی منطقی پایگاه داده است و در آن اسامی جدول‌ها و ستون‌های آنها، همچنین نوع یا دامنه ستون‌ها مشخص شده است. در شما تنها ساختار و طراحی منطقی دیده می‌شود و در آن اثری از داده‌ها نیست. داده‌های پایگاه داده در نمونه پایگاه داده دیده می‌شوند. نمونه پایگاه داده تصویری از داده‌های پایگاه داده در لحظه خاصی از زمان است.

شمای پایگاه داده مجموعه شماهای جدول‌های آن پایگاه داده است. در مثال‌های صفحات ۴ و ۵ نمونه‌های جداول course و instructor دیده می‌شود. در بالای جدول‌ها عنوان ستون‌ها نوشته شده است. این قسمت که با رنگ پس زمینه آبی مشخص شده است شمای جدول است. اگر کوچکترین تغییری در داده‌های این مثال‌ها در پایگاه داده اتفاق بیافتد نمونه دیگری از جدول (و پایگاه داده) مورد نظر خواهیم داشت.

شمای جداول و شمای پایگاه داده به روش‌های مختلفی نمایش داده می‌شوند و در این بین از نمودارها نیز استفاده می‌شود. اما معمولاً شمای جدول به شکل کلی زیر نشان داده می‌شود:

(لیست ستون‌ها) نام جدول

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

مثلاً برای جدول department که اطلاعات گروه‌های آموزشی را در خود نگه می‌دارد

شمای پایگاه داده با این صورت است:

department (dept name, building, budget)

کلیدها:

اگرچه در جدول‌های پایگاه داده در حالت عمومی می‌توانیم سطر تکراری داشته باشیم، اما به هر حال بسیار پیش می‌آید که لازم است بین سطرها تمایز قائل شویم. مثال زیر را در نظر بگیرید

جدول زیر را در نظر بگیرید	
نام دانشجو	رشته تحصیلی
صادق احمدی	کامپیوتر
بابک ناصبی	معماری
صادق بیای	هرف
رضا اکبری	کامپیوتر
علی بندری	هرف

نام دانشجو	رشته تحصیلی
صادق احمدی	کامپیوتر
بابک ناصبی	معماری
صادق بیای	هرف
رضا اکبری	کامپیوتر
علی بندری	هرف
علی بندری	هرف

حال اگر در این جدول بخواهیم یک دانشجو را حذف کنیم و آن دانشجو علی بندری باشد که از قبل در جدول بوده، حتماً به مشکل بر می‌خوریم. معمولاً دستوراتی که در پایگاه داده برای این کار نوشته می‌شوند تمام سطرهایی را که کاملاً داده‌های آنها یکی است حذف می‌کنند. هیچ نشان و علامتی وجود ندارد که مشخص کند کدام علی بندری زودتر آن یکی در جدول ذخیره شده است. روی کاغذ داده‌ها به ترتیب ورود نوشته می‌شوند اما در پایگاه داده نمایش جدول به کاربر و ترتیب قرار گرفتن سطرها در آن، بر اساس ساختمان داده‌ای است که برای ذخیره و بازیابی داده‌ها استفاده می‌شود. بنابراین ممکن است مثلاً سطر دوم جدول بعد از سطر پنجم آن ثبت شده باشد. از اینرو نمی‌توانیم از شماره سطرها برای تمایز بین آنها استفاده کنیم.

کلیدها در پایگاه داده برای تمایز چندتایی‌ها داخل یک رابطه استفاده می‌شوند. معمولاً در بین صفت‌های یک رابطه می‌توان مواردی را پیدا کرد که مقدار آنها برای هر موجودیت منحصر به فرد باشد و می‌توان از آنها به عنوان شناسه موجودیت برای تمایز از دیگر موجودیت‌ها استفاده کرد. به این صفات کلید گفته می‌شود. کلیدها انواعی دارند که در ادامه شرح داده خواهند شد.

سوپر کلید

مجموعه‌ای از یک یا چند صفت است که چندتایی واحدی را مشخص می‌کند. به عنوان مثال شمای رابطه‌ی زیر را در نظر بگیرید:

کارمند (نام، نام خانوادگی، شماره پرسنلی، کد ملی، شماره شناسنامه، تاریخ تولد، محل صدور، نام پدر، آدرس)
در این رابطه صفت‌های کد ملی و شماره پرسنلی در هر چندتایی مقدار منحصر به فردی دارند. در جدول کارمند هیچ‌وقت در هیچ نمونه‌ای از پایگاه داده، هیچ دو سطری یافت نمی‌شوند که در ستون کد ملی مقدار یکسان داشته باشند. همچنین در ستون شماره پرسنلی.

سوپر کلید می‌تواند مجموعه‌ای از صفت‌ها باشد. در ادامه تعدادی از سوپر کلیدهای جدول کارمند نوشته شده است.

(شماره پرسنلی) ، (کد ملی) ، (شماره شناسنامه، تاریخ تولد، محل صدور) ، (نام، کد ملی) ،
(نام پدر، کد ملی) ، (شماره پرسنلی، کد ملی) ، (شماره پرسنلی، نام، نام خانوادگی) ، (کد ملی، نام پدر) ،
(نام، نام خانوادگی، شماره شناسنامه، تاریخ تولد، محل صدور)

اگر K یک سوپر کلید باشد هر مجموعه از صفات رابطه که شامل K باشد یک سوپر کلید است. مثلاً در مثال بالا هر مجموعه از صفات که شامل کد ملی باشد یک سوپر کلید است. همینطور (شماره پرسنلی) و (شماره شناسنامه، تاریخ تولد، محل صدور). درست است که نام می تواند تکرار شود اما نام و کد ملی با هم تکرار شدنی نیستند. زیرا کد ملی تکرار نمی شود.

ممکن است در جدول در ستون نام خانوادگی هیچ داده ی تکراری نداشته باشیم. در این شرایط نمی توانیم نتیجه بگیریم که نام خانوادگی سوپر کلید است. سوپر کلید باید همیشه و برای همه نمونه های پایگاه داده خاصیت منحصر به فرد بودن را داشته باشد.

کلید کاندید

تنها شرط سوپر کلید بودن یک مجموعه از صفات این است که همیشه داده های آن منحصر به فرد باشد. سوپر کلید با این تعریف نمی تواند برای هدف ما که تمایز بین رکوردها است مناسب باشد. زیرا ممکن است خود حاوی چند کلید باشد مثل (شماره پرسنلی، کد ملی). از طرفی در بسیاری از سوپر کلیدها با حذف یک یا چند صفت مجموعه باقیمانده همچنان سوپر کلید است. اگر حذف صفات تا جایی ادامه پیدا کند که به کوچکترین اندازه سوپر کلید برسیم آنگاه به آن سوپر کلید مینیمم می گوییم. به عبارت دیگر سوپر کلید مینیمم سوپر کلیدی است

که با حذف هر یک از عناصر یا صفات داخل آن دیگر سوپر کلید نباشد. به سوپر کلید مینیمم کلید کاندید می‌گوییم.

در مثال قبل کلیدهای کاندید جدول کارمند عبارتند از:

(شماره پرسنلی) ، (کد ملی) ، (شماره شناسنامه، تاریخ تولد، محل صدور)

کلید اصلی

کلید کاندیدی است که طراح پایگاه داده آن را به عنوان ابزار اصلی شناسایی چندتایی‌ها، در داخل یک رابطه انتخاب کرده است. گاهی ممکن است در بین صفات یک رابطه کلید کاندید نداشته باشیم. در اینصورت معمولاً طراح پایگاه داده صفتی را که از قبل جزو صفات موجودیت نبوده به مجموعه صفات رابطه اضافه می‌کند. عموماً این صفات با نام‌های شناسه (ID)، کد (Code) و امثال آن نامگذاری می‌شوند. در شمای پایگاه داده معمولاً کلید اصلی با خطی در زیرش (underline) یا با عبارت مخفف PK مشخص می‌شود.

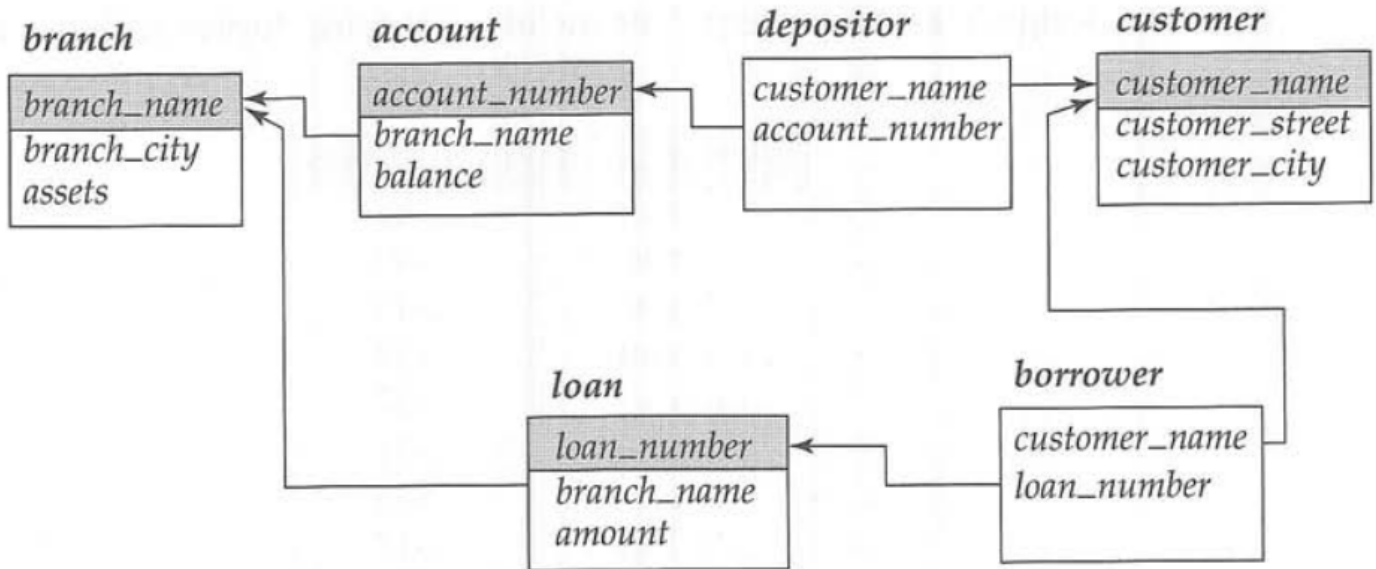
کلید خارجی

کلید خارجی صفتی از یک رابطه (مثلاً $r1$) است که در یک رابطه دیگر (مثلاً $r2$) کلید اصلی است. به رابطه‌ی شامل کلید خارجی ($r1$) رابطه‌ی مراجعه کننده (referencing relation) می‌گویند و به رابطه‌ی شامل کلید اصلی رابطه‌ی ارجاع شده (referenced relation) می‌گویند. به مثال زیر توجه کنید.

عنوان رشته تحصیلی	کد رشته ^{PK}	کد رشته تحصیلی ^{FK}	شماره دانشجویی ^{PK}	نام دانشجو
کامپیوتر	1	1	۹۷۰۰۸۱۵۲	صادق احمدی
معماری	2	2	۹۸۱۳۹۴۸۱	بابک ناصبی
برق	3	3	۹۷۱۵۷۱۲۴	صادق بیاضی
عمران	4	1	۹۴۴۴۸۱۳۲	رضا اکبری
حسابداری	5	3	۹۸۲۹۸۳۴۱	علی بندری

در این مثال کلید اصلی با PK و کلید خارجی با FK مشخص شده است. در مثال صفحه ۱۳ عنوان رشته تحصیلی در جدول دانشجو قرار داده شده بود. همانگونه که مشخص است موارد تکراری در این ستون می‌تواند بسیار زیاد باشد. اصطلاحاً می‌گوییم که این صفت افزونگی دارد. برای کنترل افزونگی (اصطلاحاً رفع افزونگی هم گفته می‌شود) عنوان رشته تحصیلی را در یک جدول دیگر قرار داده و به هر رشته یک کد منحصر به فرد اختصاص می‌دهیم. آنگاه بجای عنوان رشته تحصیلی در جدول دانشجو از کد رشته استفاده می‌کنیم. به این شکل بجای تکرار نام‌ها و رشته‌ها اعداد صحیح تکرار می‌شوند. به این کار اصطلاحاً کد گذاری گفته می‌شود. کد گذاری هم می‌تواند در مصرف حافظه صرفه جویی کند و هم می‌تواند در افزایش سرعت پردازش داده‌ها نقش داشته باشد.

شمای پایگاه داده را می‌توان با نمودار هم نشان داد در شکل صفحه بعد نمودار شمای پایگاه داده نشان داده شده است. در این نمودار وابستگی‌های کلید خارجی و کلید اصلی مشخص شده‌اند. هر رابطه به صورت یک چهارگوش نشان داده شده است که نام رابطه در بالا و صفات آن در داخل چهارگوش قرار می‌گیرند. کلیدهای اصلی با پس زمینه تیره مشخص شده‌اند. وابستگی‌های کلیدهای خارجی نیز به صورت پیکانی از صفت کلید خارجی به سمت کلید اصلی رسم می‌شوند.



تمرین‌ها

تمرین ۱: با مثال نشان دهید که چگونه کد گذاری می‌تواند باعث کم شدن مصرف حافظه و افزایش سرعت پردازش داده‌ها شود.

تمرین ۲: برای شمای پایگاه داده زیر که شامل هفت جدول است، بیست سوپر کلید بنویسید. کلیدهای کاندید، اصلی و خارجی را نیز مشخص کنید.

کارمند (شماره کارمندی، نام، نام خانوادگی، کد اداره، سن)

اداره (کد اداره، نام اداره) امانت (شماره کارمندی، شابک کتاب، تاریخ امانت)

کتاب (شابک، عنوان، کد ناشر) ناشر (کد ناشر، نام ناشر)

مؤلف (شابک کتاب، کد شخص) شخص (کد شخص، نام شخص)

جواب تمرین‌ها را به آدرس ایمیل saasghari@chmail.ir ارسال نمایید.